

AI Coding Bootcamp with Claude Code (Self-Paced)

Own the full process of planning, building, and deploying real web applications with Claude Code in this beginner-friendly bootcamp loaded with expert instruction, interactive practice, seven labs, and a capstone project.

Group classes in Live Online and onsite training is available for this course. For more information, email corporate@nobledesktop.com or visit: <https://www.nobledesktop.com/classes/ai-coding-bootcamp-with-claude-code-self-paced>



hello@nobledesktop.com • (212) 226-4149

Course Outline

Module 0: Orientation & Course Overview

- Define vite coding and how AI-assisted development differs from traditional programming
- Preview the four projects built throughout the course
- Navigate the course structure, assessments, and time commitments
- Recall foundational terms: IDE, repository, frontend, backend, API
- Complete a pre-assessment to establish a personal knowledge baseline

Module 1: How the Web Works

- Describe the request-response cycle, DNS, and HTTP
- Distinguish between frontend and backend components
- Explain what an API is and why it matters
- Identify the role of frameworks like React and Next.js
- Define key terms: DNS, HTTP, REST API, server, client, framework

Module 2: Your Tools — Getting Started with Claude Code

- Explain how Claude Code differs from chat-based AI and code-completion tools
- Install the Desktop App and configure a Claude Code session
- Use the diff review workflow to accept, reject, and comment on changes
- Create a CLAUDE.md file to give Claude project-specific context and rules
- Apply the five-component prompt structure for coding tasks
- Lab: explore a project, fix a bug, and build a small app from scratch

Module 3: Build Your First App

- Apply the three-question planning framework to scope features

- Build multi-step features including dark mode and data filtering
- Initialize Git repos, manage branches, and resolve merge conflicts
- Write symptom-based debugging prompts
- Identify key files and directories in a Next.js project
- Lab: build a professional business landing page with Next.js and Tailwind CSS

Module 4: Effective Prompting & Workflows

- Contrast outcome-based vs. implementation-based prompts
- Apply the Explore → Plan → Execute → Review workflow
- Use constraints, examples, references, and persona framing to guide AI
- Recognize when to steer a conversation vs. restart a session
- Scaffold a complete project from a single structured prompt
- Lab: scaffold and extend an original project using the four-step workflow

Module 5: Data, APIs & Backend

- Set up Supabase and connect it to a Next.js application
- Implement user authentication (signup, login, OAuth)
- Perform CRUD operations and design database schemas
- Connect to external APIs using the universal integration pattern
- Build validated forms with proper error handling
- Lab: build a full-stack app with a database, auth, and an API integration

Module 6: Design & User Experience

- Apply UX/UI fundamentals and use design references with Claude Code
- Make layouts responsive across devices with Tailwind CSS breakpoints
- Implement WCAG accessibility: semantic HTML, contrast, keyboard navigation
- Handle loading, error, empty, and success UI states
- Use browser DevTools to inspect and troubleshoot styling
- Lab: redesign an existing project with responsive layout and accessibility fixes

Module 7: Security & Privacy

- Identify the six most common web attacks: XSS, SQLi, CSRF, and more
- Apply input validation and data sanitization defenses
- Configure rate limiting, HTTPS, and environment variable management
- Apply GDPR and CCPA requirements to practical scenarios
- Audit dependencies for known vulnerabilities
- Lab: security-harden an existing project

Module 8: Ship & Scale — Deployment

- Explain key differences between development and production environments
- Deploy a Next.js app to Vercel with environment variables configured
- Connect a custom domain and configure DNS
- Apply SEO best practices: meta tags, Open Graph, Core Web Vitals
- Run the pre-launch checklist for security, performance, and accessibility
- Lab: deploy a full-stack app to a live URL

Module 9: Advanced Workflows & Power Features

- Execute coordinated multi-file changes in a single workflow
- Refactor code with AI-assisted module extraction and reorganization
- Generate test suites and practice test-driven development (TDD)
- Create custom Skills and slash commands for repetitive tasks
- Explore Hooks, Sub-Agents, MCP connectors, and the terminal CLI
- Capstone: plan, build, test, and deploy an original application

Module 10: Reference & Next Steps

- Browse an API marketplace organized by industry and category
- Access copy-paste prompt templates for common development tasks
- Reference the 50-term searchable glossary
- Identify tools, libraries, and resources for continued learning
- Chart a personal learning path beyond AI Coding Bootcamp with Claude Code